SOLOMON: Seeking the Truth Via Copying Detection

Motivation

We live in the Information Era, with access to a huge amount of information from a variety of data sources. However, data sources are of different qualities, often providing conflicting, out-of-date and incomplete data. Data sources can also easily copy data from others, propagating erroneous data. Thus, identifying high quality information and sources is non-trivial.

We demonstrate SOLOMON, whose core is a module that detects copying between sources. We demonstrate that we can effectively detect copying, leverage the results in truth discovery, and provide a user-friendly interface to assist users in identifying sources that best suit their information needs.

Data set: Data extracted by searching computer science books from an online bookstore (source) aggregator, AbeBooks, in 2007 [4].

Back-End

The back-end of SOLOMON takes the data from various sources and the schema mappings as input, performs data fusion and infers quality measures of sources, copying relationships between sources, and true values for each data item.

Copying Detection: Copying detection proceeds in two steps [1,2]. The first step, local detection, discovers copying for each pair of sources independently of other sources. The second step, global detection, identifies co-copying and transitive copying.

Truth Discovery: When deciding the truth from conflicting values, SOLOMON not only considers which value the majority of sources vote for, but also ignores the copied values and gives higher weight to data from more accurate sources [1].

Quality Measurement: SOLOMON measures the quality of sources by completeness (the percentage of data that are provided), accuracy (correctness of the provided values), consistency (number of distinct values provided for each item), and redundancy (number of records provided for each object) [2].

Front-End

The front-end of SOLOMON provides a search and browsing interface to the user, generating visualizations and explanations on users’ demand. The web-based front-end is written in JavaScript and thus completely portable.

Decision Explanation: A user often wonders not only “what” but also “why”. SOLOMON provides explanation of various decisions, interpreting the underlying Bayesian analysis in a way that non-technical users can understand.

Visualization: SOLOMON provides an effective visualization to assist understanding of source quality and copying relationship. It applies the GMap techniques [3] and shows the sources in a map where closely related (by copying) sources are put close to each other. It also provides visualization for explanation of various decisions.

References